SOEST 04-03 c.2

Use of Neural Networks with Advection-Diffusion-Reaction Models to Estimate Large-Scale Movements of Skipjack Tuna from Tagging Data

M. Shiham Adam & John R. Sibert

SOEST 04-03
JIMAR Contribution 04-350

Pelagic Fisheries Research Program

SH
214.6
.S64
no.04-03
c.2

NMFS Honolulu Library
00000000002277

SH
214.6
.S64
no.04-03

Use of Neural Networks with Advection-Diffusion-Reaction Models to Estimate Large-Scale Movements of Skipjack Tuna from Tagging Data

M. Shiham Adam*

and

John R. Sibert

Pelagic Fisheries Research Program
University of Hawaii at Manoa
Honolulu, HI 96822

*Marine Research Centre, Ministry of Fisheries,
Agriculture and Marine Resources
H. White Waves, Malé, Republic of Maldives

LIBRARY
Pacific Islands Fisheries Science Center
NOAA Fisheries
2570 Dole St.
Honolulu, HI 96822-2396

SOEST 04-03
JIMAR Contribution 04-350

LIBRARY
Pacific Islands Fisheries Science Center
NOAA Fisheries
2570 Dole St.
Honolulu, HI 96822-2396

Adam, M. S., and J. R. Sibert, Use of Neural Networks with Advection-Diffusion-Reaction Models to Estimate Large-Scale Movements of Skipjack Tuna from Tagging Data, 34 pp, University of Hawaii, Joint Institute for Marine and Atmospheric Research, 1000 Pope Road, Honolulu, HI 96822, 2004.

Contents

1 Motivation 2

2 The model 2

3 Parameterizing movement fields 4

3.1 User of neural networks 5

3.2 Scaling the output 7

3.3 Constraining the parameters 7

4 Integrating the neural networks and estimation of the model parameters 10

5 Input data 13

6 Results and discussion 15

6.1 Simulation 21

6.2 Results from a typical fit 23

7 Recommendations for further research 29

8 Acknowledgments 30

9 References 30

Abstract

We explored integration of multi–layer, feed–forward neural networks into an advection–diffusion–reaction model to estimate large–scale tuna movement from conventional tagging data. Two neural networks were used to compute spatially and temporally varying movement parameters from a series of input data specific to the computational grid. The first neural network computed the zonal and meridional components of the advection term, and the second computed the diffusion term. The input data, which are considered proxies for habitat that may be important for determining skipjack behavior, include temporal (calendar year, month, Southern Oscillation Index), spatial (latitude, longitude, water depth, seabed topography, distance from the land), and spatio–temporal data (sea surface temperature and surface currents). The computed movement parameters from the neural networks were used in the advection–diffusion–reaction model, along with the tag-attrition parameters (natural and fishing mortality rates), to compute predicted tag recaptures. The training of the neural network and simultaneous estimation of the other model parameters were achieved by minimizing the negative log of a likelihood function that relates the observed and predicted tag recoveries.

The method was applied to the skipjack tuna data from the central and western Pacific Ocean. The results were compared against an earlier analysis of the same data where movement parameters were mapped onto a 10-region $\times$ 2-season advection-diffusion-reaction model. Numerous neural network configurations and parameter combinations were investigated, and three-layer neural networks with 4-6 input nodes in the input layer and with 3-5 nodes in the hidden layer were found to be reasonable. None of the models gave adequate estimates of the neural network (movement) parameters. In all the cases the movement parameters of the advection-diffusion-reaction model were poorly defined by the neural network models. The attrition parameters were estimated satisfactorily in all the models, and the predicted and observed tag recaptures showed good agreement, particularly for sporadic and isolated recaptures. This is in contrast with the regional and seasonal parameterization where model prediction was better for short–term recoveries.

Problems using neural networks in this way are discussed, and possible improvements in the neural network configuration and alternative parameterization of the movement parameters are suggested.

1 Motivation

Tunas have ocean–wide distributions in the tropical and subtropical regions of the world. They are highly mobile and are classified as 'highly migratory' in many international conventions. Within their broad geographic distribution, and in all the oceans, a multitude of fisheries exploits the resource, both on the high seas and in the coastal regions. Most of the coastal tuna fishing nations are small island countries and their economies are often dependent on the tuna resource. Despite the tuna's economic importance and their potential for fishery interactions, the effects of their movements on fishery management, at a level of individual fishery area, are normally neglected.

The state–of–the–art, ocean–wide stock assessment models that are in use today consider the movement within the context of 'bulk transfer' models (*Fournier et al.* 1998; *Hampton and Fournier* 2001). Although these models have been successful in predicting the change in population density in broad geographic regions, they cannot be used to predict change in an arbitrary point (or at fishery levels) because they are not continuous in space.

Advection–diffusion–reaction (ADR) models of continuous time and space have been successfully used to model tagged populations of skipjack tuna on large scales (*Sibert et al.* 1999; *Adam and Sibert* 2002; *Sibert and Hampton* 2003). The models estimate movement parameters using likelihood methods. However, in the absence of theory explicitly relating movement behavior to the environment, structural assumptions that associate movement parameters within regions and seasons were made for estimating the movement parameters.

In this investigation, we attempted to take a step further to improve upon the movement parameterization by making use of neural networks to compute local movement parameters from a variety of habitat proxies. The spatio-temporally varying movement parameters were computed by the neural network and used in the ADR model described by *Sibert et al.* (1999) to simultaneously estimate the parameters for both the neural network and the ADR model by maximum likelihood. We illustrate the approach using the SSAP[1] data set used by *Sibert et al.* (1999).

2 The model

A tuna tagging experiment begins with release of tagged fish from one or more locations. Subsequent releases may occur and the experiment continues until the last tagged fish (or simply tags) have been recovered. The basic information collected include release and recapture times and their geographic positions. The number of tags, N_{xyt}, at any given time, t, at geographic position, (x, y), is given by

$$N_{xyt} = \sum_{c}^{c_t} \tilde{N}_{xytc}, \tag{1}$$

[1]Skipjack Survey and Assessment Programme, Secretariat of the Pacific Community, Noumea, New Caledonia

where $\tilde{N}_{xytc}$ is the number of tags remaining alive at time, t, at the geographic position (x, y) that were released from cohorts, $c = 1, ..., c = c_t$. In the case where a subsequent cohort, d, release occurs at time t_d at location (x_d, y_d) the density of that location is reassigned to the sum of its present value and the cohort value, i.e.,

$$N_{xy}^{t} \leftarrow N_{x_d y_d t_d c_d} + \sum_{c=1}^{c_t} \tilde{N}_{xytc}. \tag{2}$$

Assuming the behavior of cohorts are independent, and that complete mixing of tagged and non–tagged fish occurs following release such that they have equal probability of recapture, the population of tags may be represented by the ADR model (*Bills and Sibert* 1997; *Sibert et al.* 1999)

$$\frac{\partial N}{\partial t} = \frac{\partial}{\partial x}\left(D\frac{\partial N}{\partial x}\right) + \frac{\partial}{\partial y}\left(D\frac{\partial N}{\partial y}\right) - \frac{\partial}{\partial x}(uN) - \frac{\partial}{\partial y}(vN) - ZN. \tag{3}$$

The model resolves the local rate of change in the density of tags into random (or diffusion, D), directed (or advective, u, v) and reaction (or mortality Z) components. Details of the boundary conditions and issues relating to the solution of the PDE are given in *Bills and Sibert* (1997) and *Sibert et al.* (1999). In this exercise we only used reflective boundaries and a fully implicit upwind differencing method of solving the PDE, which guarantees numerical stability even with large rates of movement (see *Press et al.* (1993) for a discussion on implicit differencing). Wthout further constraints, movement parameters (u, v, D) will vary over time and space. The initial condition for N is

$$N_{xyt_0} = \begin{cases} \tilde{N}_{x_c y_c c_0 t_0} & \text{over release locations} \\ 0 & \text{elsewhere.} \end{cases} \tag{4}$$

The reaction term includes components of natural mortality, M, and fishing mortality, F. Using the convention

$$Z_{xyt} = M + \sum_{f} F_{xyft}, \qquad \text{where} \qquad F_{xyft} = Q_f E_{xyft}. \tag{5}$$

Fishing mortality of the fleet f is then assumed to be a constant proportion, Q_f (the catchability coefficient) of the fishing effort E_{xyft}. The predicted number of recaptured tags, $\hat{C}_{xyt}$, is given by

$$\hat{C}_{xyt} = \beta_f \frac{F_{xyft}}{Z_{xyt}}(1 - e^{-Z_{xyt}})N_{xyt} \tag{6}$$

where β_f is the fishery–specific tag reporting rate, independently estimated from tag-seeding experiments (e.g., *Hampton* 1997).

3 Parameterizing movement fields

Without further information to parameterize movement behavior, the usual approach in the past has been to constrain movement parameters to be constant in 'space–time blocks' of the model domain (*Sibert et al.* 1999; *Adam and Sibert* 2002). The idea here is to make the assumption that movement behavior would be the same in regions and seasons so as to estimate the regional and seasonal movement parameters. Those analyses suggest that movement information in the tagging data were highly variable and might be better explained by continuously varying movement parameters. Also, the region–season parametrization often produces arbitrary discontinuities in population density that are not biologically realistic. An alternative parameterization of the ADR model in time–space discretized form was used by *Kleiber and Hampton* (1994). Using skipjack tuna tagging data from the Solomon Islands, they modeled the movement (in this case transfer rate coefficients from the grid boundaries) as functions of the number of FADs[2] and the proportion of the island mass that falls within the grid. While successful in detecting a FAD and an island mass effect, these approaches would not be applicable for open ocean where such features would not be available. *Lehodey et al.* (1998, 2003) have shown, in a simulation model, that a proxy for tuna forage can be used to parameterize movement parameters in an ADR model that produces realistic distribution of skipjack tuna.

In principle, it would be possible to parameterize the movement fields using a mechanistic model based upon the scientific understanding about the relationship between the movement behaviour and the explanatory variables. However, these relationships are currently unknown. It is generally accepted that the small–scale distribution and abundance of skipjack tuna is affected by the the availability of forage (*Lehodey et al.* 1997). The forage comprises epipelagic prey items ranging from large zooplankton, such as euphausiids, amphipods and other small crustaceans, to bait fish such as oceanic anchovy (*Lehodey et al.* 1998). The distribution and abundance of the forage are regulated by oceanographic processes, such as seasonal upwelling events driven by oceanic features that occur at scales of few kilometers and months. However, direct measurements of forage density at resolutions compatiable with spatio-temporal scales of the tagging data are currently are not available. Similarly, indirect measurements of forage, such as ocean color, that are obtainable by means of remote sensing have not been satisfactory for the existing tagging data.

Earlier efforts at parametrizing the movement fields to a simple 'habitat index' based on sea-surface temperature and sea-surface current were not unsuccessful (*Sibert and Bertignac*, unpublished analysis). The problem appeared to be lack of detail (interpolated climatology data) in the data sets available at the time. However, recent advances in computer simulations of skipjack populations from ADR models have been encouraging. For example, *Lehodey et al.* (2003) showed that skipjack tuna catch history is reproducible at ocean basin level by constraining the movement fields to primary production, sea surface temperature, and oceanic currents that were simulated from a coupled physical–biogeochemical model.

[2]Fish Aggregating Devices, floating objects specifically deployed to attract fish

3.1 Use of neural networks

Neural networks are proving to be attractive devices for modeling complex nonlinear processes without making an *a priori* relationship between the predictor and the explanatory variables (*Lek and Guégan* 2000). The December 2001 special issue of *Ecological Modelling* (Vol. 146, 1-3) contains several applications of artificial neural network modeling. Neural networks have been used in many fishery applications; forecasting fish recruitment (*Chen and Ware* 1999), estimating fish density (*Brosse and Lek* 2002) and standardizing fishing effort data (*Maunder and Hinton* submitted).

Our approach was to use neural networks to compute local movement parameters of an ADR model using a series of habitat proxies as input data. The parameters of both neural net and ADR model are estimated simultaneously from a maximum likelihood algorithm. Before explaining how the neural network is integrated into the ADR model, we describe below the type of the neural network used and various constraints we imposed on the neural net parameters.

We used simple multi–layer, feed–foward neural networks. A typical configuration of neural network is shown in Figure 1. The processing elements, called nodes (or neurons in the jargon of neural networks), are arranged in layers. The first layer, called the input layer, takes in the explanatory variables (input variables) and output comes off from the last layer, called the output layer. In between the input and ouput layers is the hidden layer(s). For any given number of input nodes, the model complexity may be increased by increasing the hidden layers and/or increasing the number of nodes in the hidden layer(s). For the purposes of this analysis we used three-layered networks only. Each node is connected to all other nodes of adjacent layers and receives input and sends output through these connections. In the hidden layers the nodes simply sum the weighted values of the input variables, adds a bias term, and passes the value through a scaling function (or activation function in the neural net jargon). A three–layered neural network may mathematically be represented by

$$
\begin{aligned}
h_k &= \phi_k\left(\sum_{j=1}^{n}(w_{jk}x_j + b_k)\right) \qquad k = 1,..,k = m, \\
o_t &= \theta_t\left(\sum_{k=1}^{m}(w_{kt}h_k + b_t)\right) \qquad t = 1 \text{ or } 2
\end{aligned} \tag{7}
$$

where h_k is the hidden–layer output, which is scaled by the activation function $\phi(x)$ (see below). The first subscript in the weights, w, indexes the nodes in the preceding layer and the second indexes the nodes in the current layer, x_j, where $j = 1,..j = n$ are the input values of the input layer, b is the bias term, and m and n are the number of the input–layer nodes and hidden–layer nodes respectively. The hidden–layer output becomes the input values for the output layer, o_t, which uses the output scaling function $\theta(x)$ (see below) to scale the output values to appropriate range for use in the ADR model. The activation function, $o(x)$, can be sigmoid or linear. We chose a sigmoid function that scales the values between -1 and +1 as given by

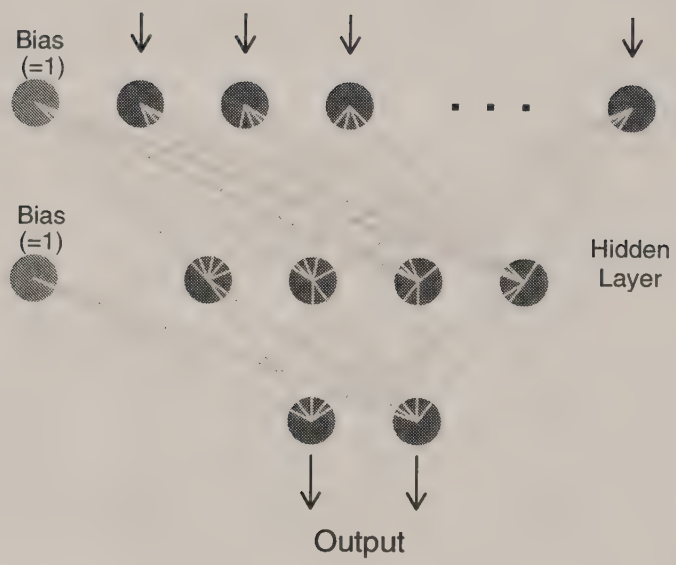


Figure 1: A feed–forward neural network with single hidden layer and with two output nodes in the output layer.

$$\phi(x) = \frac{2}{\pi}\arctan(x). \tag{8}$$

3.2 Scaling the output

To be able to use the neural network output as movement parameters of the ADR model, the output values have to be scaled to the appropriate range of the ADR models' movement parameters. Previous analyses suggested that D would be in the range of 0–50,000 square nautical miles per month and u and v around -30 to 30 nautical miles per month. To map the neural network output to these ranges of the movement parameters, we used a flexible scaling function that takes in the desired range of the parameter along with a 'slope' and an 'offset' parameter. The scaling function is given by

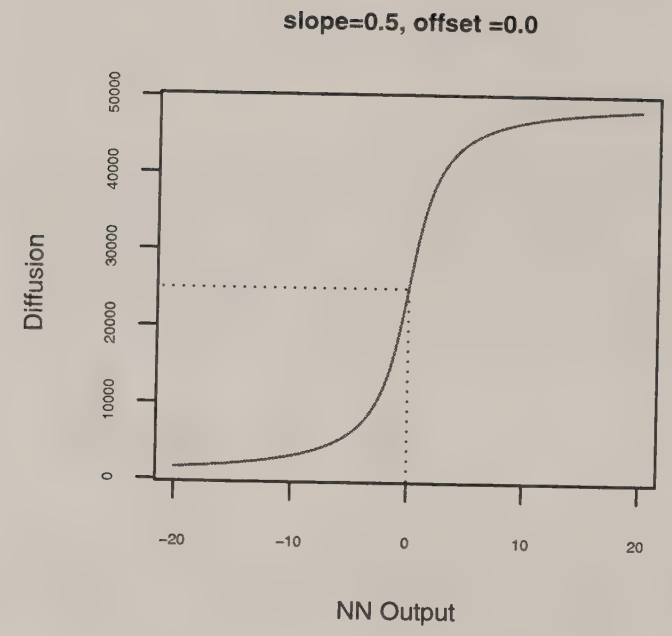
$$\theta(x) = \min + (\max - \min)\left(\frac{\arctan(\alpha * [x - \beta])}{\pi + 0.5}\right) \tag{9}$$

where α and β are slope and offset parameters respectively. Higher values of α increase the 'sigmoid nature' of the function and very high values tend to make it a step function. Conversely, decreasing α makes the function more linear. The parameter β is simply the offset of the input values relative to zero. An example of scaling functions with different values of α and β is shown in Figure 2.

While it may be possible to estimate the values of α and β from the tagging data itself, we found that the parameters are confounded with the weights connecting from the hidden layer to the output layer. For example, the scaled output for a neural network with two nodes in the hidden layer, ignoring the constant terms, is given by $\alpha\{\arctan(w_1h_1 + w_2h_2 + b - \beta)\}$. Any given value of this expression may be obtained by dividing and multiplying the parameters with a constant, for example $\frac{\alpha}{2}\{\arctan([2w_1h_1] + [2w_2h_2] + [2b] - [2\beta])\}$, giving multiple solutions. Similarly by adding and subtracting a constant to b and β would have no effect on the value of the expression. We therefore made α and β constants in the model fitting procedure. Alternatively, a linear scaling function may be used that would avoid the use of these parameters.

3.3 Constraining the parameters

In a highly generalized nonlinear model structure like the nueral network, the issues of local and multiple global minima are a pervasive problem (*Sussmann* 1992; *Intrator and Intrator* 2001). In the conventional 'training' of the neural network (model fitting), a subset of the data, called the validation set, is left aside for testing the performance of the model (method of cross validation). During training, the error (the sum of the squared differences between the observed and predicted) of the training and the validation set is compared after each function evaluation over the entire training set (called an epoch in the neural net jargon). If the training and validation set were independent and representative of each other, the decrease in error of the training and validation set as a function of increasing epochs should be more or less similar (*Lek and Guégan* 2000). Optimum training for a given model is



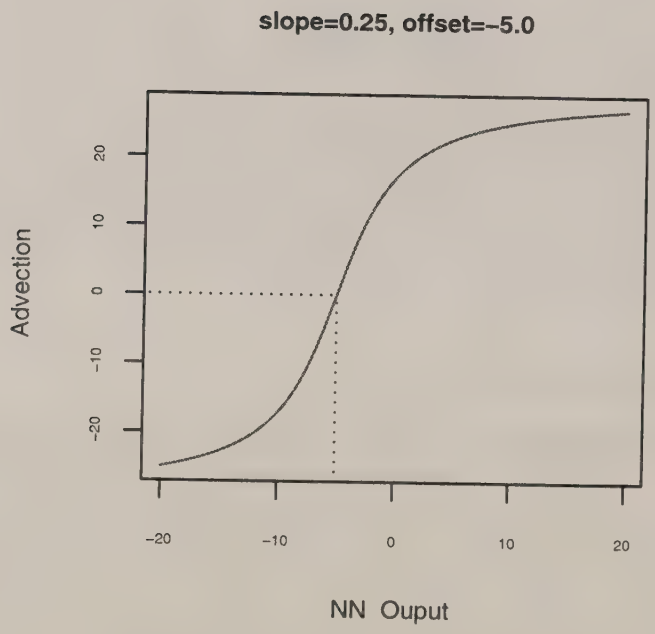


Figure 2: Scaling function for diffusion and advection with offset (α) and slope (β) parameters.

achieved when the error of the validation set starts to increase even if the error from the training set continues to decrease. This discrepancy is due to over-fitting (over-generalizing) of the model. The idea here is to train the neural net to a degree so that the general feature of the underlying nonlinear model of the data is captured in the parameters of the neural net. The fitting procedure does not guarantee that the network has found a minimum or indeed a unique solution in parameter space. Thus, the success of the training procedure and the resulting neural network's ability to accurately predict future events depends upon inclusiveness of the observation data and the correct choice of neural network configuration, the latter found by trial and error.

For our purposes of estimating the maximum likelihood of the data given a particular model, the fitting procedure should find the global minimum of the error surface in parameter space, thus converging to a unique solution. In the neural net models, it has been shown that non-uniqueness occurs with certain neural net configurations (*Sussman* 1992; *Albertini et al.* 1993). To avoid such undesirable effects we constrained the weights connecting from the hidden layer to the output layer to be positive and be in ascending order. For example if $w_{ij},\ i = 1, .., 4; k = 1, 2$ represents the matrix of weights connecting from a neural network with four nodes in the hidden layer connecting to two output nodes in the output layer, we constrained the weights to be the difference between the weights of the current node and the previous node. For example,

$$w_{11} < w_{21} < w_{31} < w_{41}, \qquad w_{i1} > 0$$

$$w_{12} < w_{22} < w_{32} < w_{42} \qquad w_{i2} > 0.$$

Nondecreasing weights were achieved by re–parameterizing the weights as

$$w'_{11} = w_{11};\ \ w'_{21} = w_{21} + w_{11};\ \ w'_{31} = w_{31} + w_{21} + w_{11};\ \ w'_{41} = w_{41} + w_{31} + w_{21} + w_{11}$$

so that $w_{31} = w'_{31} - w_{21} - w_{11}$ where w'_{31} is the actual neural network parameter to be estimated. Simple simulation experiments with the above constraints on the hidden–layer weights consistently gave unique solutions. In addition to the hidden–layer weight constraints we also penalized the likelihood with a value dependent on the total magnitude of the weights connecting from the input layer to the hidden layer (*Hsieh* 2004). If the likelihood function is $f(\theta; \mathbf{X})$ this is achieved by

$$\ell(\mathbf{X}|\theta) = f(\theta; \mathbf{X}) + P \sum_{j=1}^{n} (w_{jk})^2$$

where P is a weight magnitude penalty parameter. Larger values of P would tend to make the weights smaller in magnitude, yielding a smoother and less nonlinear solution than when P is small or zero.

At the start of the model–fitting procedure, the weights of the neural network were initialized with small random values and the bias terms initialized with zero. The attrition parameters were initialized with values close to the estimated values of the regional and seasonal movement model estimates.

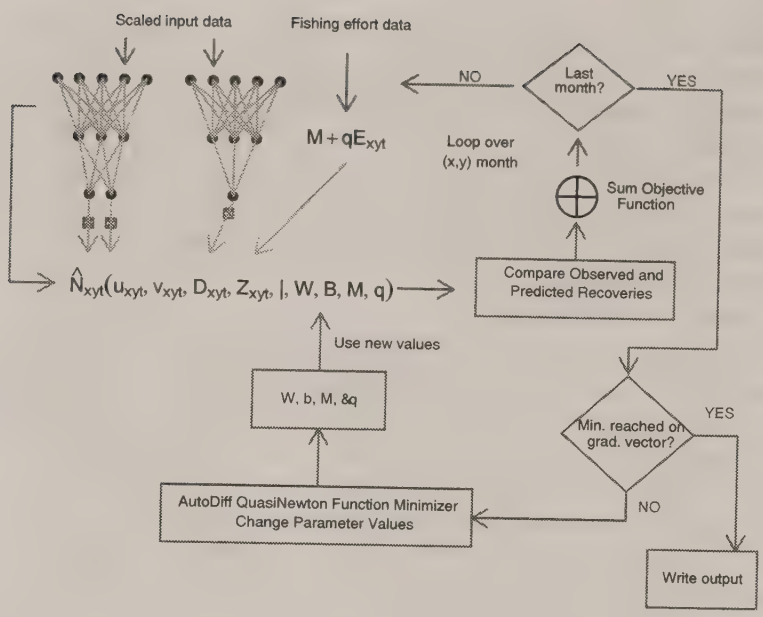


Figure 3: Schematic flowchart showing the main features in the integration of a neural network into the ADR model.

4 Integrating the neural networks and estimation of the model parameters

We used two neural networks: one that calculates two output values corresponding to the u and v, the other calculating a single output value corresponding to the D. At each time step and at each computational element in the model, the neural networks calculate three output values corresponding to the three movement parameters of the ADR model from a series of grid–specific input data (see below). These values are then scaled using the appropriate scaling functions of the movement parameters and passed to the ADR model. The movement parameters along with the attrition parameters (q and M) are used to calculate local density of the tags. A flow chart representing the integration of neural networks to the ADR model and the parameter estimation procedure is given in Figure 3.

In order to estimate the parameters of the model, we assumed that recaptures occur in each grid cell, and in each time step follow a Poisson probability distribution. The likelihood of the data given the model is (*Sibert et al.* 1999)

$$\ell(\mathbf{X}|\theta) = \prod_{xyft} \left(\frac{e^{-\widehat{C}^t_{xyf}} \widehat{C}^t_{xyf}{}^{C^t_{xyf}}}{C^t_{xyf}!} \right). \tag{10}$$

The estimates of the parameters were found by minimizing the negative log of the likelihood function across all observations. We used the Autodiff Library's quasi–Newton function

minimizer (*Otter Research* 2002), which makes use of the gradient of the partial derivatives of the model parameters. Analytically correct partial derivatives were computed using adjoint code based on the concept of reverse mode automatic differentiation (*Griewank and Corliss* 1991). Bounds are imposed on all estimated parameters to assure stability of the numerical function minimization algorithm (*Bard* 1974).

The bounds set on the parameters are listed in Table 1. The numerical function minimizer iterates until the maximum of the absolue value of the gradient components is less than a convergence criterion, normally $\max_i |\frac{\partial \ell}{\partial \theta_i}| \leq 0.001$.

Table 1: Parameter ranges set in the estimating procedure.

Parameter	Lower bound	Upper bound
Input–layer weights	-10.0	+10.0
Hidden–layer weights	0.0	40.0
Bias terms	-100.0	+100.0
M	0.0	0.5
q	0.0	1.0

5 Input data

The parameters of the ADR model have meaning in the real world. In the neural network, however, the parameters (weights and biases) do not have meaning in terms of the process being modeled. Thus, neural networks are often referred to as black boxes. It is therefore important that we choose candidate input variables that are related to the process being modeled.

Some of the input data fields we used vary only spatially, others temporally, and others vary both spatially and temporally. For example latitude and longitude fields vary only spatially while the year and month and Southern Oscillation Index (SOI) vary only temporally. Oceanographic data fields vary both spatially and temporally. Following is a complete list of the input variables we considered in the analysis.

- Temporal fields
  - **Calendar year and month:** We used the year and month of the models' time horizon to provide time information. A smooth sigmoid transformation for the month signal was used to avoid discontinuity when month change occurs at the end of each year.
  - **Southern Oscillation Index:** Standard SOI[3] index. This field gives a time signal since the SOI was the monthly average across the Pacific Ocean. In the central and western Pacific Ocean the SOI has a profound influence on the east–west distribution of the skipjack abundance (*Lehodey et al.* 1997).
- Spatial fields
  - **Geographic location:** We used the latitude and longitude positions of the grid element to provide the spatial information to the model. Both would give a smooth signal in the x and y dimensions of the model.
  - **Water depth:** We used average water depth at computational grid, calculated using the ETOPO5[4] to give spatial heterogeneity to the model.
  - **Topography:**[4] This field was taken to be the difference between the maximum (at sea level) and minimum elevation at the computational element and provide spatial heterogeneity.
  - **Distance from land:** Calculated as the nearest distance from the computational element to the land, visible in the model resolution. This information may be useful as a proxy to the so called 'island mass effect' (*Kleiber and Hampton* 1994).
- Combined Temporal and Spatial fields[5]
  - **Zonal and meridional surface current:** Meridional and zonal sea surface current in the computational element.
  - **Sea surface temperature:** Meridional and zonal sea–surface temperatures in the computational element.
  - **Simulated 'tuna forage':** Simulated forage field from a biogeochemical model.
  - **Simulated habitat index:** 'Habitat' field predicted by parameterizing the forage field.

All input data fields were scaled between -1 and +1. Figure 4 shows water depth and topography fields. Histograms of the SST, and the zonal and meridional currents for the SSAP region are shown in Figure 5.

[3]http://tao.atmos.washington.edu/data_sets/globalsstenso/, accessed December 2001

[4]ETOPO5 (Earth Topography - 5 Minute), National Geophysical Data Center (NGDC), Boulder, CO.

[5]Provided by Dr. Patrick Lehodey, Secretariat of the Pacific Community, Noumea, New Caledonia. The data are the same as those used in *Lehodey* (2001).

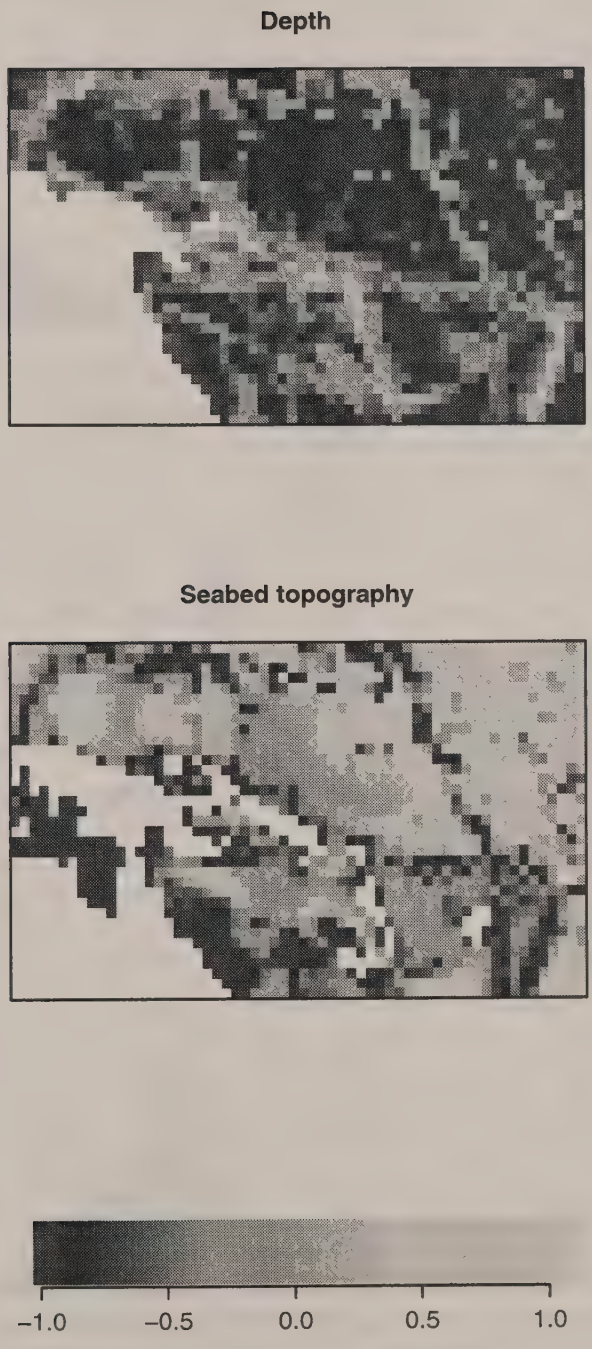


Figure 4: Scaled spatial fields of the water depth and the topography

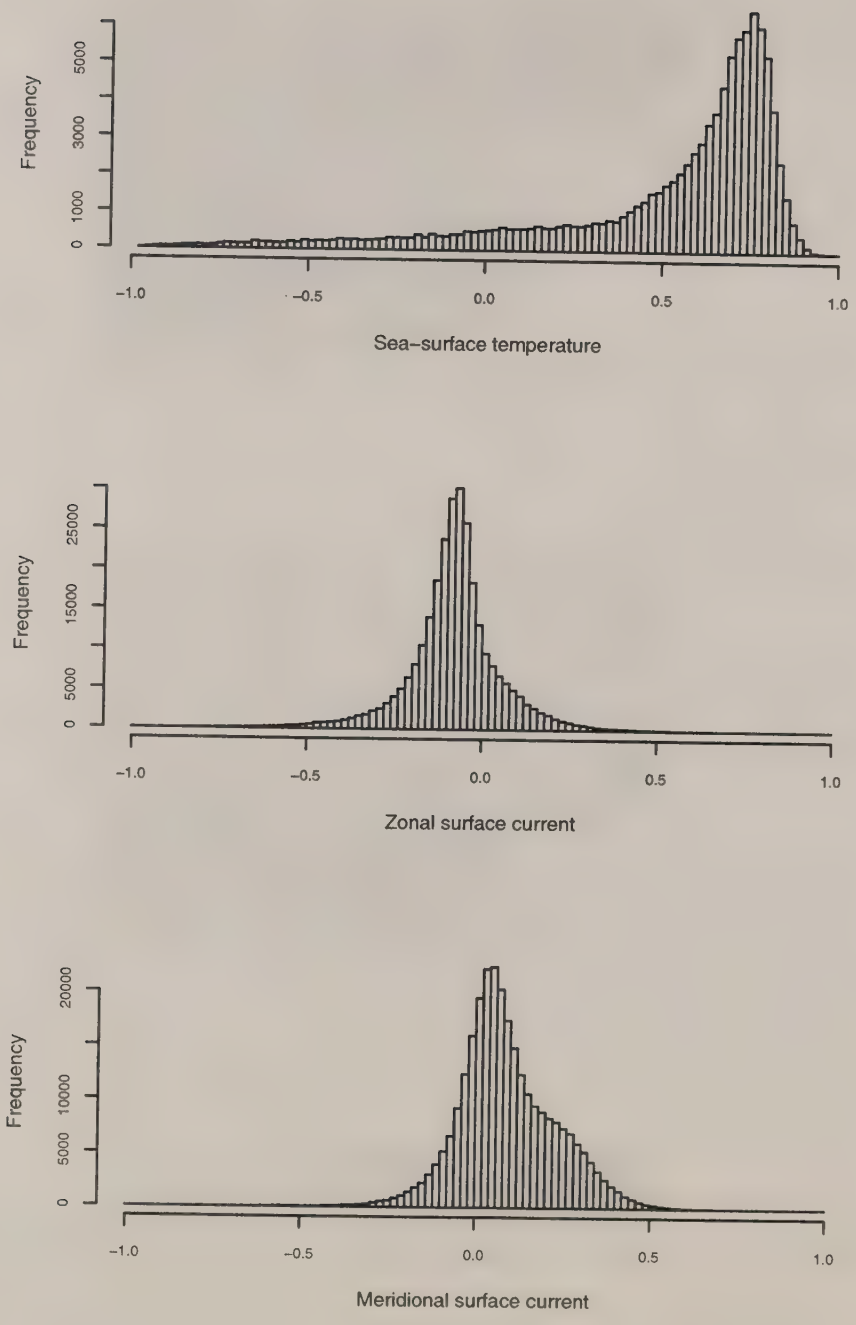


Figure 5: Frequency distributions of the sea surface temperature (SST), zonal surface current (USC) and meridional surface current (VSC).

6 Results and discussion

For brevity and convenience of discussion the models are represented by 15 alphanumeric characters. For example, the model UV642B6-D531B4-MQ (66 estimated parameters, $n = 66$) represents a UV–neural network with 6 input layer nodes, 4 hidden layer nodes, 2 output, and 6 bias terms, followed by the D-neural network configuration. The last two letters represent the natural mortality and the catchability coefficients. Flexible coding of the model allowed testing of a multitude of neural network configurations and parameter settings. After a series of trials, a reasonable neural network configuration was found to be models of 4–6 input variables (= input–layer nodes) with 3–5 hidden nodes in the hidden layer.

Table 2 shows estimates of the attrition parameters and the objective function values for a series of neural network configurations. The parameter estimates of the 10-region × 2-season model selected by *Sibert et al.* (1999) as the best fit, referred here as the 'benchmark' fit, are also given for comparison. None of the model fits showed satisfactory convergence. In rare cases where convergence occurred, the hidden–layer weights and the bias terms of the D–neural network were always on the bounds. Increasing the range of the weights was found to be unhelpful. A typical model fitting (training) history is shown in Figure 6. In most models that we tested, the likelihood function ceased to decrease after 500–600 function evaluations despite continued increase in iterations (to an upper limit of 2000). In those cases virtually all of the hidden–layer weights and the bias terms of the D–network were on the bounds. For this reason a 'stopping criterion' based on the rate of decline in likelihood function (until the value of the function did not decrease by 0.001 consecutively for 20 evaluations) was used.

Despite the problems in estimating the neural network parameters, the attrition parameters were estimated reasonably well in all the models we tested (Table 2). The estimates of the natural mortality rate were slightly lower compared with the benchmark fit. Except for the Japanese purse seine and possibly Solomon Islands pole–and–line, the catchability coefficients in all the fleets were nearly the same as in the benchmark fit.

The pattern of predicted diffusion field often showed close correlation with the spatial data. For example, in the neural networks that used the depth and topography field, the estimated diffusion field showed high resemblance (see Fig. 14) with little change across the time steps. A reason for this may be the presence of prominent 'ridge features' that change input values abruptly (cf. Fig. 4). Compared with the spatial input data, the temporal input data appeared to have little influence on the movement fields.

The neural network parameter estimates were very sensitive to the initial values of the weights. Although repeating a fit with the same initial values consistently gave the same parameter estimates, different starting values gave vastly different estimates of the parameters (Figure 7). Possible reasons for the sensitivity to starting values are that the data are uninformative with respect to the neural network parameters or that there are multiple maxima in the likelihood surface. Scatter plots of the movement parameters estimated from the benchmark fit against the average input data values for those space–time strata showed that the relationship, if any existed, was weak (Figure 8 and Figure 9). Similarly, the relationships with the depth and topography data were also poor.

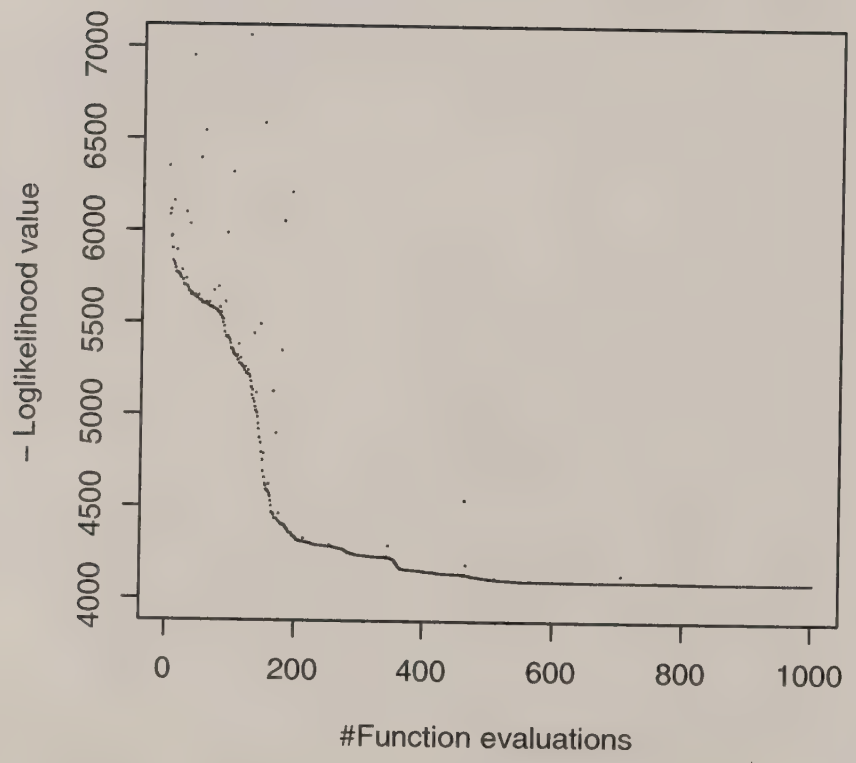


Figure 6: A typical training history of the model. In most cases the function values cease to decline after about 500–600 iterations despite increase in the number of function evaluations.

Table 2: A sample of the estimates of attrition parameters from different neural network configurations. The estimates of the 'benchmark' fit from the regional–seasonal paramterization is given for comparision.

Model configuration	M	q_{pgpl}	q_{sbpl}	q_{fgpl}	q_{jppl}	q_{jpps}	npar	-LogLike	| G |	Notes
10 region x 2 season	0.111	0.00041	0.00041	0.00457	0.00081	0.00561	66	4007.68	0.0003	Benchmark fit
1:UV641B5-D632B5-MQ	0.083	0.00043	0.00064	0.00427	0.00078	0.00346	68	4059.71	5.0553	Does not converge
2:UV732B5-D731B4-MQ	0.072	0.00053	0.00074	0.00503	0.00083	0.00275	66	4682.39	6.5274	Does not converge
3:UV732B5-D731B4-MQ	0.088	0.00039	0.00054	0.00451	0.00099	0.00466	66	4086.93	110.6232	Does not converge
4:UV632B5-D631B4-MQ	0.076	0.00045	0.00056	0.00471	0.00084	0.00378	60	4058.12	1.3113	Does not converge
5:UV632B5-D631B4-MQ	0.085	0.00044	0.00059	0.00461	0.00096	0.00299	60	4078.59	15.9012	Does not converge
6:UV632B5-D531B4-MQ	0.091	0.00048	0.00073	0.00432	0.00091	0.00277	57	4212.59	0.0073	D weigts on bounds

Neural network input data for above:
1:UV641B5-D632B5-MQ: UV: mon, lat, lon, zdt, soi, frg; D: mon, lat, lon, zdt, soi, frg
2:UV732B5-D731B4-MQ: UV: yrr, mon, lat, lon, soi, sst, hab; D: yrr, mon, lat, lon, soi, sst, hab
3:UV732B5-D731B4-MQ: UV: yrr, mon, lat, lon, soi, sst, hab; D: yrr, mon, lat, lon, soi, sst, hab
4:UV632B5-D631B4-MQ: UV: yrr, mon, lat, lon, zdf soi D: yrr, mon, lat, lon, zdf soi
5:UV632B5-D631B4-MQ: UV: mon, lon, lat, zdt, zdf, soi D: mon, lon, lat, zdt, soi, sst
6:UV632B5-D531B4-MQ: UV: yrr, mon, lat, lon, zdt, soi D: mon, lon, lat, zdt, soi

Notes:
1: Benchmark fit with region × season model
2: Does not converge
3: D weights on bounds

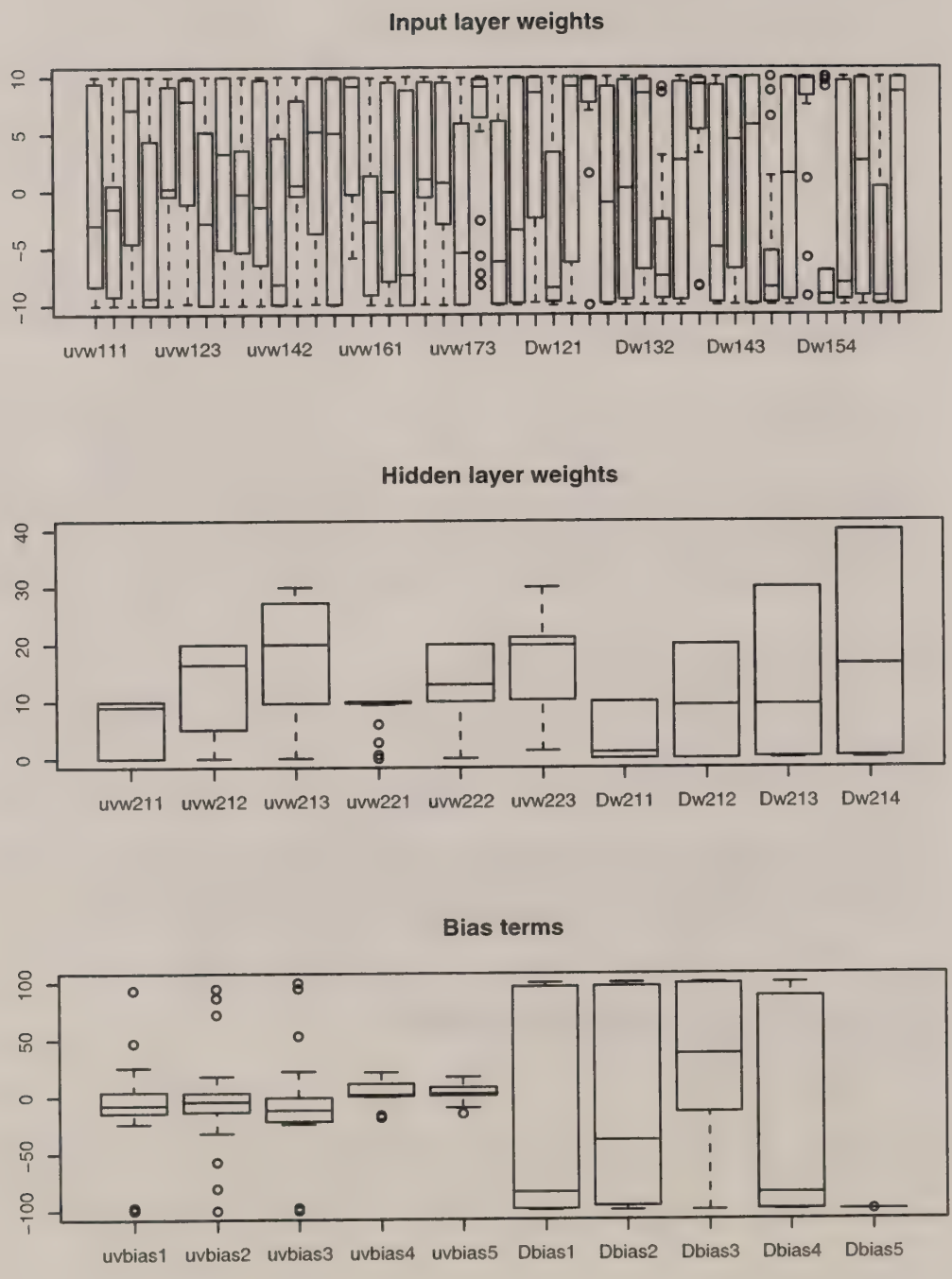


Figure 7: Distributions of estimates of the neural network parameters for a UV732B5-D731B5-MQ ($n = 66$) model using 20 different sets of initial values of weights and bias terms.

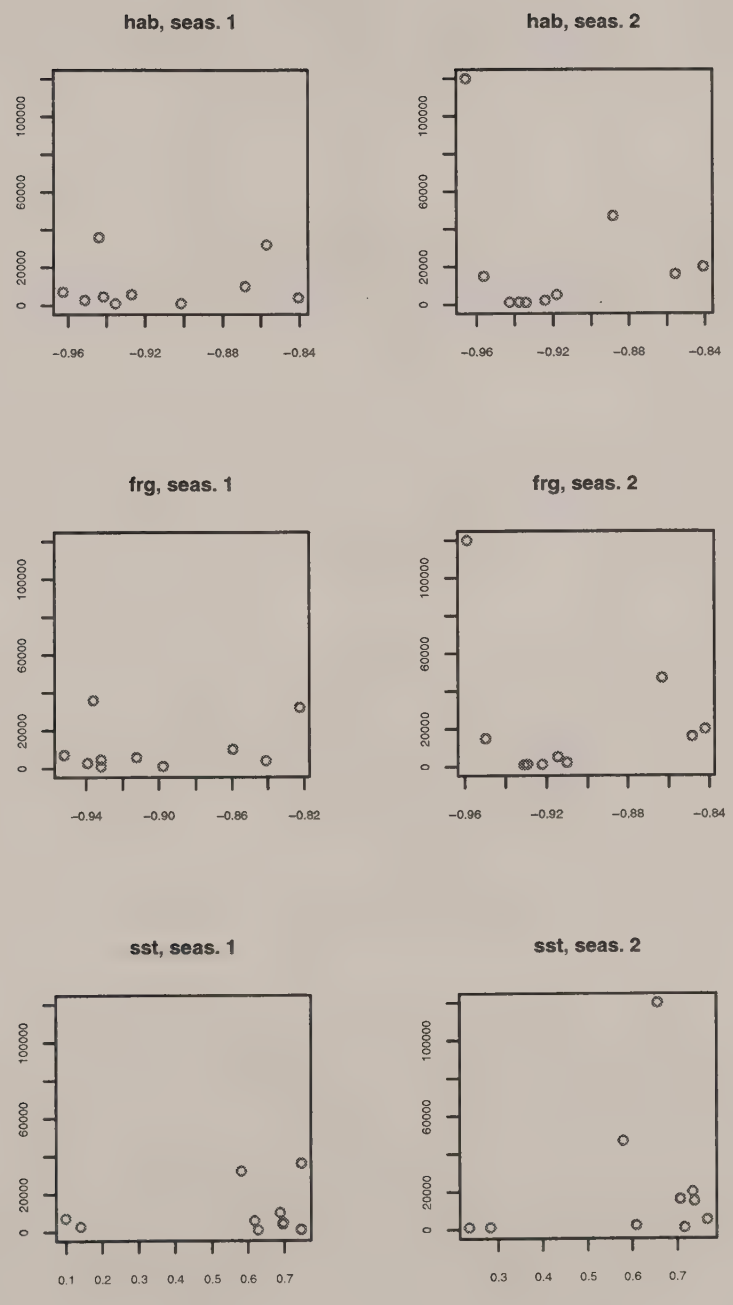


Figure 8: Relationship between the benchmark fit estimated values of D against the average values of spatio–temporally varying neural network input data (habitat (hab), forage (frg), and sea surface temperature (sst)). Left panels are for season 1 and right panels are for season 2.

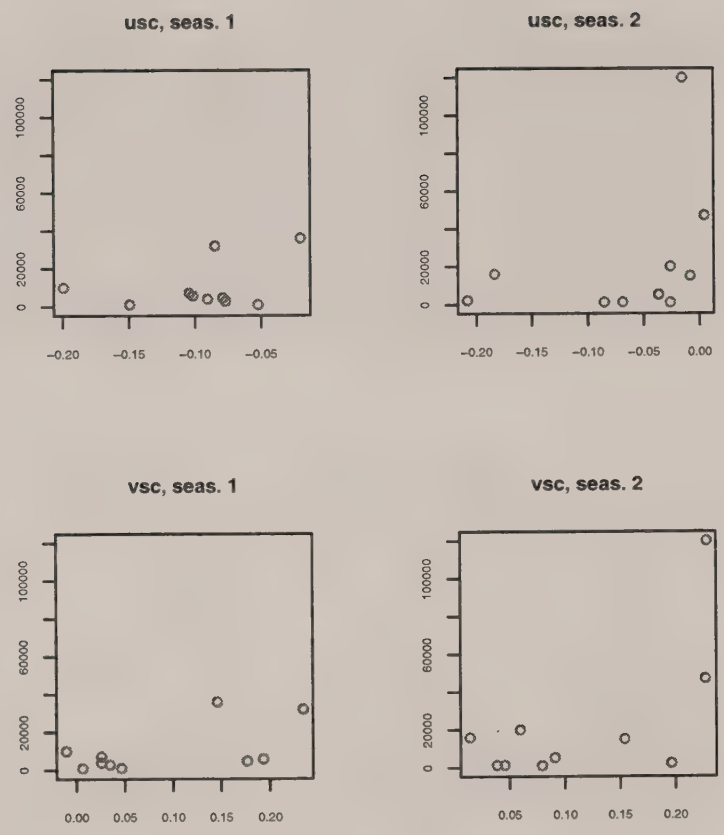


Figure 9: Relationship between the benchmark fit estimated values of D against the average values of spatio–temporally varying neural network input data (zonal (usc) and meridional (vsc) surface current). Left panels are for season 1 and right panels for season 2.

In all the cases, the advection field was strongly oriented to a northwesterly direction at the eastern boundary of Australia. Incidently there was no release and hardly any recapture in this area of the model (see Fig. 12). The spare and sporadic nature of the recapture data may also explain the multimodal distribution of the estimates of diffusion (see Fig. 13).

6.1 Simulation

In order to investigate the reliability of the estimating procedure, we simulated a series of recapture data sets from the model (UV732B5-D731B4-MQ, $n = 66$) using an estimated set of parameter values. In the simulation, movements were deterministic but recaptures were sampled from a probability distribution where the predicted tag density from the ADR model was considered to be the expected value of a Poisson random variable. Using different starting values for the weights and bias terms at each time, the parameters were estimated for 50 realizations of tag–recapture data. The bias in the estimates, calculated as the difference between the true and the estimated value divided by the standard deviation of the estimates, is shown in Figure 10. The estimates of the weights show large bias, some even larger than two standard deviations indicated by the horizontal lines in the figures. The attrition parameters, however, were estimated with reasonable accuracy similar to the average estimates (cf. Table 2).

A plausible reason for this behavior of the neural network model could be related to the nature of the recapture observations in the data, but perhaps more importantly it is due to the paucity of the recapture data in the model. In the conventional tagging data the movement path of the fish is not known; only the release and recapture locations are known. For a given time–at–liberty, the fish could have moved in a multitude of paths to be recaptured at the observed time–grid strata. The implicit assumption in the ADR model is that large numbers of releases and recaptures should, at the population level, reveal the population movement that would be statistically estimable from the ADR model. It is thus possible that multiple maxima in the likelihood surface for a set of tag recaptures to have originated from more than one possible movement path between points of release and recapture.

In reality, the recaptures in any grid–time strata are very few. In fact, of the 99,180 (58 months $\times$ 1710 non-land grids) computational grids in the model domain, only 952 (0.95%) grids had observed recaptures; 99% of the model domain was void of recapture data for which the parameters of the model would have to explain the movement (or more precisely the continuously changing gradient field) of the density of the tags. In the regional $\times$ seasonal parameterization of the ADR model, the accuracy of parameter estimates is higher when there are larger numbers of recaptures in a region $\times$ season stratum (*Bills and Sibert* 1997). In practice, regions are adjusted so that each region $\times$ season stratum contained at least one tag recapture. In the regional $\times$ seasonal parameterization, the movement field was effectively constrained to be uniform in each region $\times$ season strata, whereas in the neural network model, the movement field may vary continuously across the entire model domain at $1° \times 1°$ resolution. Given the paucity of recaptures relative to the region $\times$ season parameterization, it is therefore not surprising that we were not able to accurately estimate the movement parameters.

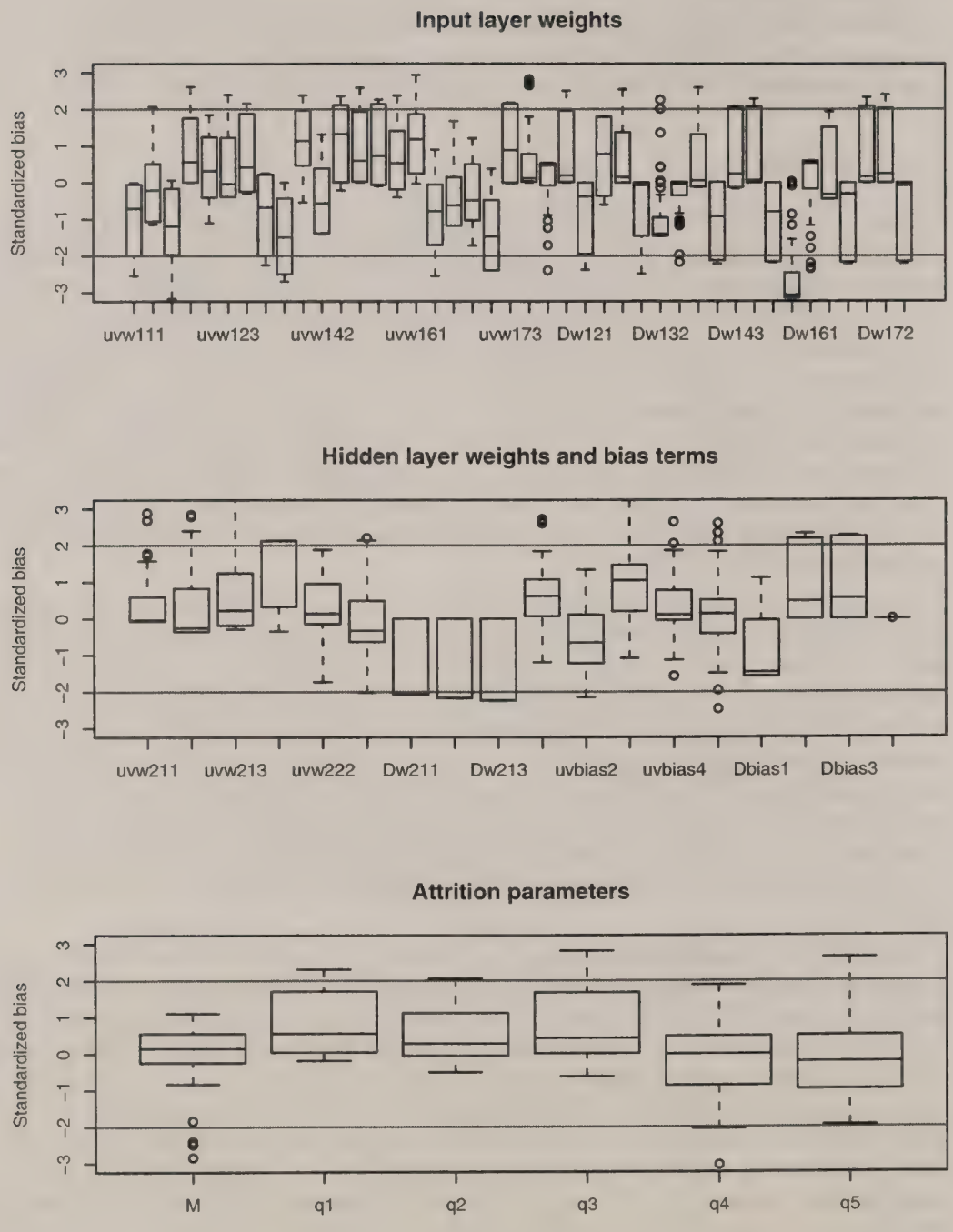


Figure 10: Bias in the estimates of the neural networks parameeers from simulated data (UV732B5-D731B4-MQ, $n = 66$). The bias is calculated as the difference between the 'true' value and the estimated value divided by the standard deviations of the estimates.

It is not clear why the behaviour of the D-neural network is different from the UV-neural work. Part of the reason might be that D is the variance of mean positions at times that are represented by the advection parameters (*Sibert and Fournier* 2001). With very few recaptures in the model domain one would expect the variance of the mean positions to be unusually large. Alternatively it is possible that the input data we used were not suitable proxies for the variables that regulate the movement behavior of skipjack tuna. Lack of apparent correlation of the estimated regional and seasonal movements with the mean habitat proxies were mentioned earlier.

The model resolution to which the movement field is calculated was $1° \times 1°$ (i.e., $\approx$ 12,000 square kilometers) in monthly time steps. However, habitat features critical in regulating the skipjack behaviour may operate on a much finer space–time resolution. For example, zooplankton swarms on which skipjack are known to forage occur at scales of around 100 km at time scales of weeks to a month (*Dickey* 1991), approximately the same scale as the model resolution. The habitat data aggregated to the model resolution may not be at an appropriate scale to yield movement information in the tag recapture data. The problem is exacerbated both by the low coverage and declining quantity of the recapture data over time.

Advective and diffusive parameters both vary at the scale of the computational elements in the model. The fine scale variation in (u, v) may confound the variability in D. In other words, advective gradients may introduce sufficient variability in movement to move the tags through the model domain without recourse to gradients in diffusion. Such confounding would be exacerbated by the 'numerical viscosity' introduced by upwind differencing (*O'Brien* 1986).

6.2 Results from a typical fit

Despite the estimates of the neural network parameters that were on the bounds, nearly all the fits showed reasonably good agreement with observed and predicted tag recaptures. The observed and predicted tag recoveries from UV632B5-D641B5-MQ ($n = 68$), using neural networks with year, month, latitude, longitude, topography, and SOI as input data for both neural networks, and a linear output scaling function is shown in Figure 11. The top panel is the aggregated tag returns over the life of each tag release cohort summed over the model area, and the bottom panel is the observed and predicted tag returns over time course of the experiment summed over the model area. The predicted recoveries from the benchmark fit are also given for comparison. Compared with the benchmark fit, the neural network parameterization better predicted recaptures at longer durations of time–at–liberty, but was worse at predicting the recaptures at shorter times–at–liberty. This was opposite to the results obtained for the regional $\times$ seasonal parameterization. Figure 12 shows tag returns aggregated over the computational grids in the entire time dimension of the model. There was generally good agreement between the observed and predicted tag returns in each grid. The agreement was particularly good for the long–term, sporadic, and isolated recaptures.

The distributions of the movement parameters obtained for this fit are given in Figure 13. The distributions of u and v are of typical in nearly all the fits. The distribution of the

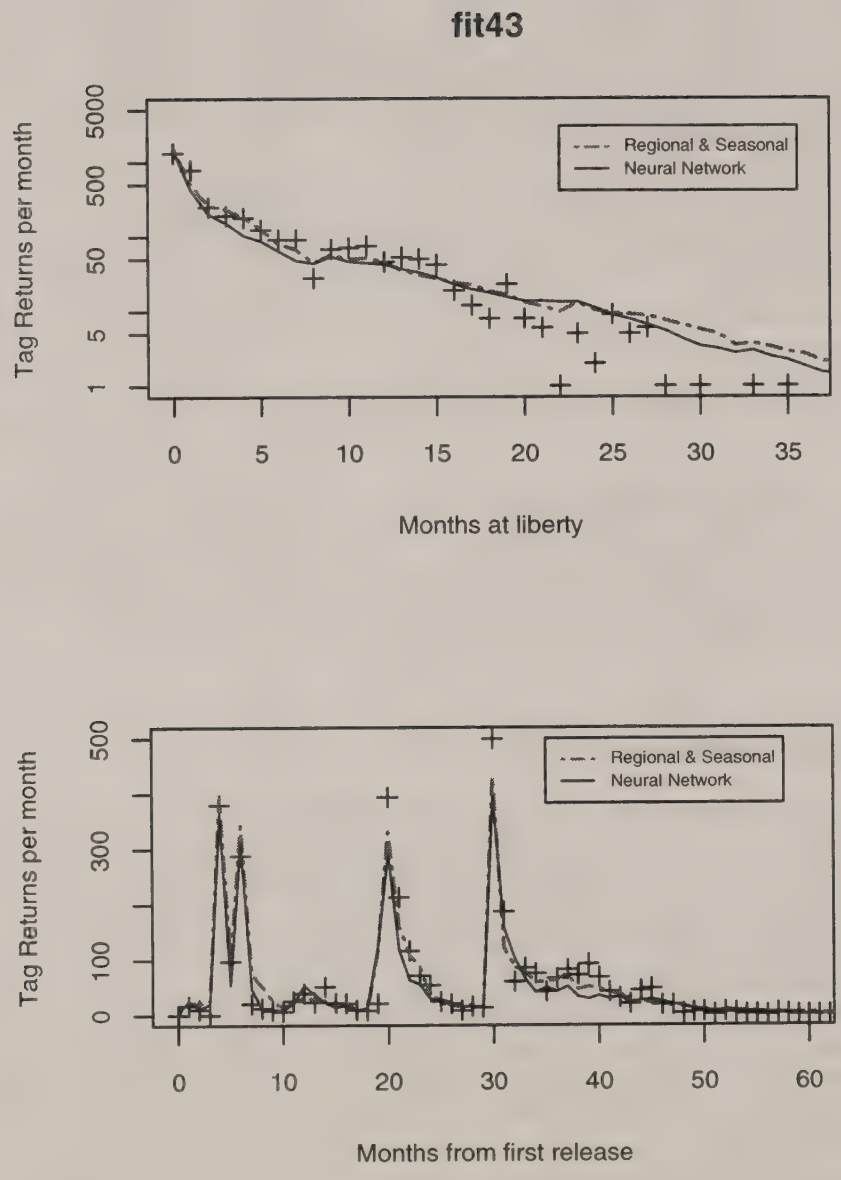


Figure 11: A typical fit of the neural network model (UV632B5-D641B5-MQ, $n = 68$) The benchmark fit, $n = 66$, is overlaid for comparison.

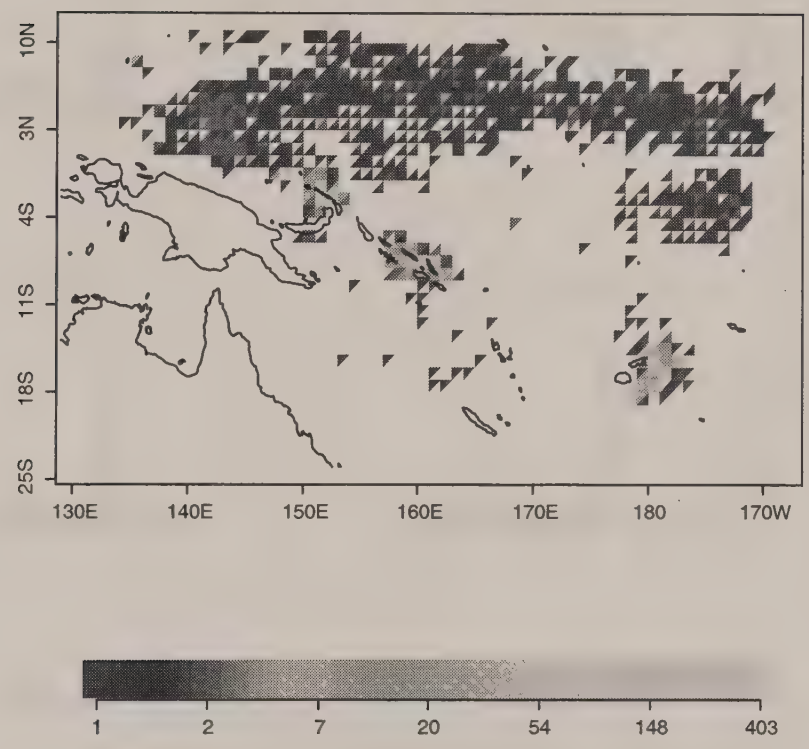


Figure 12: Spatial distribution of observed and predicted tag returns over time for model (UV632B5-D641B5-MQ, $n = 68$). Each one-degree square of the model region is divided into two triangles, the upper left triangle representing the observed tag recaptures and the lower right representing the predicted tag recaptures. The color of the triangle represents the number of returns on the logarithmic scale. Two triangles forming a square of uniform color indicate close agreement between the observed and the predicted recaptures in one–degree geographic areas.

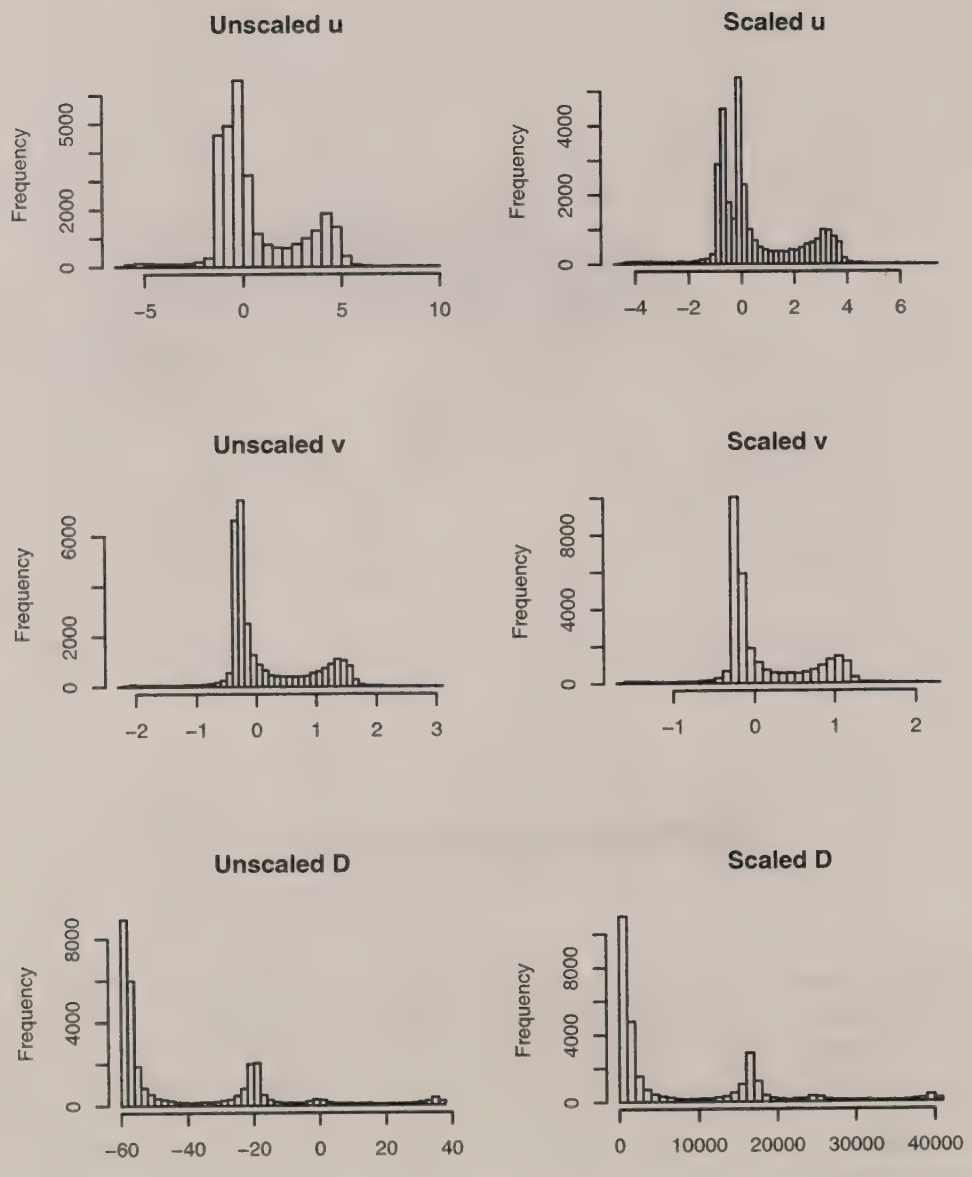


Figure 13: Distributions of the u, v, and D (fit43; UV632B5-D641B5-MQ). The figures on the left panels are unscaled values, i.e, the neural network output, and the panels on the right are scaled values, using a linear scaling function.

D, however, was highly variable between the fits. Most of them tended to show multi–modal distributions as were estimated in this fit. Given that the model fit did not converge, it will be difficult to provide meaningful interpretations of these distributions. A sample of the movement fields showing the density of the tags and the advection and diffusion parameters is given in Figure 14.

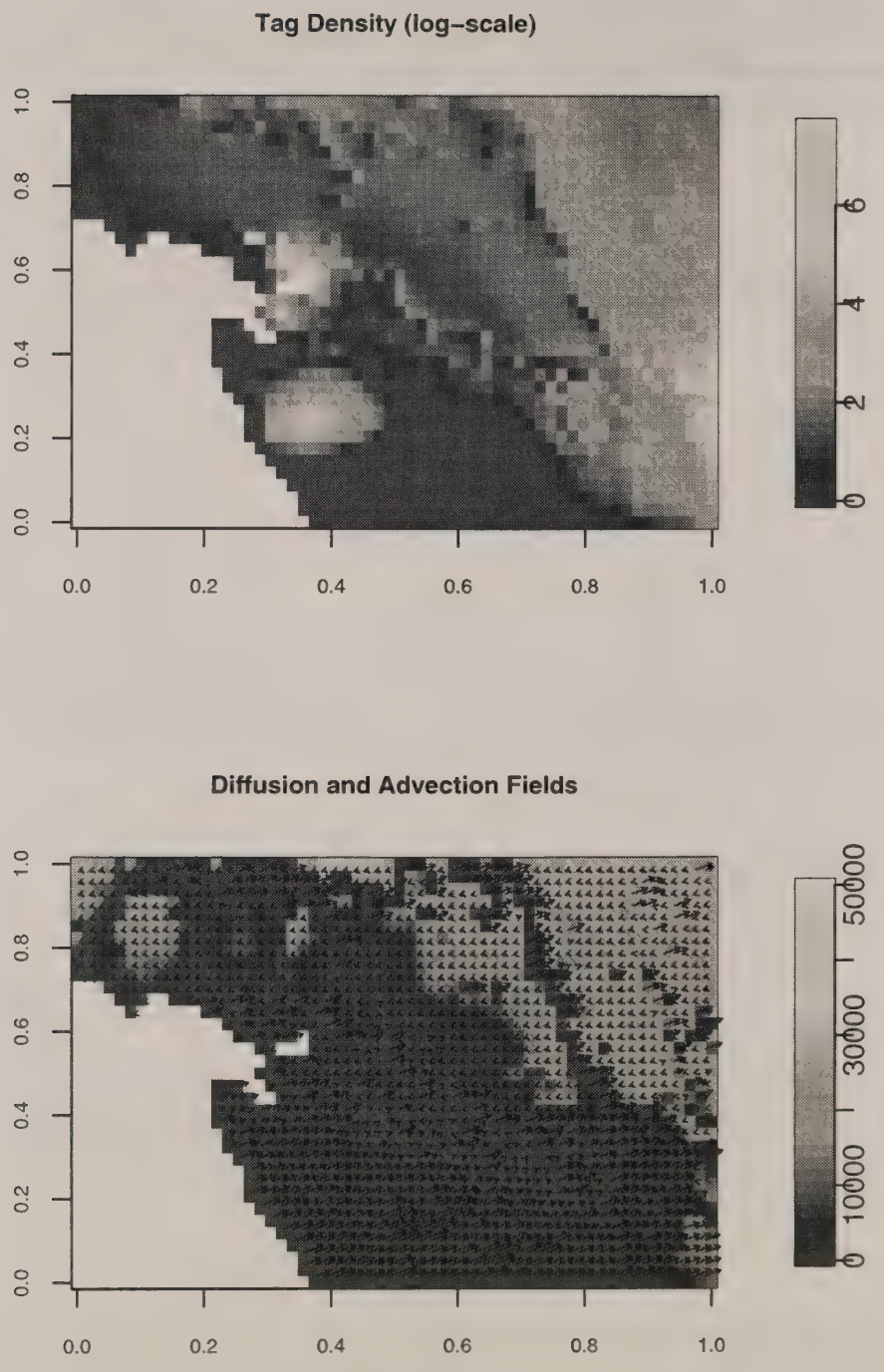


Figure 14: Sample tag density and movement fields from UV632B5-D641B5-MQ.

7 Recommendations for further research

- Change the parameterization to eliminate the gradients in D. Either esimate a single uniform value of D or set its value at an arbitrary constant.

- The recapture data were too few in quantity to estimate the continuously varying movement parameters. To compensate for this lack of recapture data, the approach followed in the regional $\times$ seasonal movement parameterization may be used. Movement parameters, for example, may be computed over 5-degree $\times$ 2 month time steps instead of the 1-degree $\times$ 1-month resolution that was used for this analysis. This would allow for more effective use of the sparse movement information in the recapture data. Implementing this scheme would require only minor modification of the existing model code.

- Future attempts should also explore use of Generalized Additive Models (GAMs) in place of the neural networks for computing local movement parameters. Somewhat similar to the neural networks, GAMs are highly flexible statistical models that are often used to identify and characterize nonlinear regression effects.

- Alternative parameterization of movement patterns: The time-space discretized approach of ADR model implemented by *Kleiber and Hampton* (1994) considers movements as fluxes, or transfer coefficients, across the cell boundaries. In their approach the flux rates are modified by the presence of FADs and island masses, where FAD effect is computed from a model.

  Using a similar approach, the computed value of the neural network may be used to modify the transfer coefficients as follows:

$$\begin{aligned}
\mathbf{m}^{\uparrow} &= \mathbf{D}.\mathrm{NN}(\mathbf{X};\theta) \\
\mathbf{m}^{\downarrow} &= \mathbf{D}.\mathrm{NN}(\mathbf{X};\theta) \\
\mathbf{m}^{\rightarrow} &= \mathbf{D}.\mathrm{NN}(\mathbf{X};\theta) \\
\mathbf{m}^{\leftarrow} &= \mathbf{D}.\mathrm{NN}(\mathbf{X};\theta)
\end{aligned}$$

  Here $\mathrm{NN}(\mathbf{X};\theta)$ is the neural network model, where $\mathbf{X}$ are the input data and θ the model parameters. The output of the neural network may be directly used to avoid the output scaling functions. Lower values of the neural network output indicate lack of local features that affect movement behavior, while higher values affect the behavior more strongly. Note that in this parameterization $\mathbf{D}$ is referred to as diffusivity and is different from diffusion, D, of the continuous form of the ADR model (see Appendix A of *Kleiber and Hampton* (1994) for a numerical treatment of the difference in the two approaches). Also there is no explicit advection term. Lack of habitat features (i.e., lower neural network output) indicates diffusivity is uniform. But uniform dispersion would be the exception rather than the norm for tunas.

8 Acknowledgments

The study was funded by the cooperative agreement NA67RJ0154 from the National Oceanic and Atmospheric Administration. The views expressed are those of the authors and do not represent views of the NOAA or its affiliated agencies or institutions. Suggestions made by Anders Nielsen on neural network parameter constraints and help with R scripts are greatly appreciated. Johnoel Ancheta's advice on C++ programming and development and maintenance of the inhouse graphics display package, 'Akira', is also appreciated.

9 References

Adam, M. S., and J. R. Sibert. 2002. Population dynamics and movements of skipjack tuna (*Katsuwonus pelamis*) in the Maldivian fishery: analysis of tagging data from an advection-diffusion-reaction model. *Aquat. Living Res.* 15: 13-23.

Albertini, F., E. D. Sontag, and V. Malliot 1993. Uniqueness of weights for neural networks, In pages 115–125, R. Mammone (ed.) *Artificial Neural Networks for Speech and Vision.* Chapman and Hall, London.

Bard, Y. 1974. *Nonlinear Parameter Estimation.* Academic Press. San Diego, CA. 341 pp.

Bills, P. J., and J. R. Sibert. 1997. Design of tag-recapture experiments for estimating yellowfin tuna stock dynamics, mortality and fishery interactions. Pelagic Fisheries Research Program, Honolulu, Hawaii.

Brosse, S., and S. Lek. 2002. Relationships between environmental characteristics and the density of age-0 Eurasian perch *Perca fluvatilis* in the littoral zone of a lake: a nonlinear approach. *Trans. Amer. Fish. Soc.* 131: 1033-1043.

Chen, D. G. and D. M. Ware. 1999. A neural network model for forecasting fish stock recruitment. *Can. J. Fish. Aquat. Sci.* 56: 2385-2396.

Dickey, T. P. 1991. The emergence of concurrent high resolution physical and bio–optical measurements in the upper ocean and their applications. *Rev. Geophys.* 29(3): 383-413.

Fournier, D. A., J. Hampton, and J. Sibert. 1998. MULTIFAN–CL: a length based, age–structured model for fisheries stock assessment, with application to South Pacific albacore, *Thunnus alalunga. Can. J. Fish. Aquat. Sci.* 55: 2105–2116.

Griewank, A., and G. F. Corliss. 1991. *Automatic differentiation of algorithms: theory, practice and application.* SIAM, Philadelphia, PA.

Hampton, J. 1997. Estimation of tag reporting and tag-shedding rates in a large–scale tuna tagging experiment in the western tropical Pacific Ocean. *Fish. Bull.* 95: 68-79.

Hampton, J. and D. Fournier, 2001. A spatially disaggregated, length–based, age–structured population model yellowfin tun *Thunnus albacares* in the western and central Pacific Ocean. *Mar. Freshwater Res.*, 52:937–963.

Hsieh, W. 2004. Nonlinear multivariate and time series analysis by neural network method. *Rev. Geophys.* 10.1029/2002RG000112 42(1) RG1003.

Intrator, O, and N. Intrator. 2001. Interpreting neural–network results: a simulation study. *Comput. Stat. Data Anal.* 27(3): 373–393.

Kleiber, P., and J. Hampton. 1994. Modeling effects of FADs and islands on movement of skipjack tuna (*Katsuwonus pelamis)*: Estimating parameters from tagging data. *Can. J. Fish. Aquat. Sci.* 51:2642–2653.

Lehodey, P., M. Bertignac, J. Hampton, and A. Lewis. 1997. El Nino Southern Oscillation and tuna in the western Pacific. *Nature* 389: 715-718.

Lehodey, P., J. M. Andre, M. Bertignac, J. Hampton, A. Stoens, C. Menkes, L. Memery, and N. Grima. 1998. Predicting the tuna forage in the equatorial Pacific using a coupled dynamical bio-geochemical model. *Fish. Oceanogr.* 7(3/4):317–325.

Lehodey, P. 2001. The pelagic ecosystem of the tropical Pacific Ocean: dynamic spatial modeling and biological consequences of ENSO. *Progr. Oceanogr.* 49: 439-468.

Lehodey, P., F. Chai, and J. Hampton. 2003. Modeling climate–related variability of tuna populations from a coupled ocean–biogeochemical–populations dynamics model. *Fish. Oceanogr.* 12(4/5):483–494.

Lek. S. and J.F. Guégan, 2000. *Artificial Neuronal Networks. Application to Ecology and Evolution.* Springer–Verlag, Berlin. 262 p.

Maunder, M. and M. Hinton. Estimating relative abundance from catch and effort data using neural networks. Submitted to *Can. J. Fish. Aquat. Sci.*

O'Brien, J. J. 1986. The hyperbolic problem, In O'Brien, J. J. (ed.), *Advanced Physical Oceanographic Numerical Modeling*, D. Reidel, Dordrecht.

Otter Research Limited, 2002. http://otter-rsch.com/, accessed December 2002.

Press, W. H., B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. 1993. *Numerical Recipes in C: The Art of Scientific Computing.* 2nd ed., Cambridge University Press, Cambridge. 1020 p.

Sibert, J., and D. Fournier. 2001. Possible models for combining tracking data with conventional tagging data. In pages 443–456, Sibert, J. and J. Nielsen (eds.) *Electronic Tagging and Tracking in Marine Fisheries Review: Methods and Technology in Fish Biology and Fisheries.* Dordrecht: Kluwer Academic Press.

Sibert, J., J. Hampton, D. A. Fournier, and P. J. Bills. 1999. An advection-diffusion-reaction model for the estimation of fish movement parameters from tagging data, with application to skipjack tuna (*Katsuwonus pelamis*). *Can. J. Fish. Aquat. Sci.* 56: 925-938.

Sibert, J., and J. Hampton, 2003. Mobility of tropical tunas and the implications for fisheries management. *Marine Policy.* 27:87–95.

Sussmann, H. 1992. Uniqueness of the weights for minimal feed forward nets with given input–output map. *Neural Network* 5:589-8-593.

Western Pacific
Regional Fishery
Management Council

University
of Hawaii
at Manoa